Design and Simulation of PID Controller for a Self-Balancing Robot

PART A

1- Model your self-balancing robot as an inverted pendulum system. You can use the standard dynamic equation of the inverted pendulum as your starting point:

$$I\ddot{\theta} = mgL\theta + \tau$$

where,

- θ = angle of tilt from vertical (radians)
- m = mass of the pendulum (kg)
- L = distance from pivot to center of mass (m)
- I = moment of inertia about the pivot (kg·m²)
- τ = input torque
- g = acceleration due to gravity (9.81 m/s²)

2- Measure the physical parameters of your robot required for this model and calculate the moment of inertia (I).

3- Build a Simulink model of the system using integrator, sum, gain, and PID controller blocks. Use your measured parameters for the system constants and gains.

4- Tune the PID parameters to achieve stable balancing with minimal overshoot and steady-state error. You may use Simulink's PID tuner or adjust manually based on your simulation results.

5- Apply a step disturbance (e.g. a torque impulse) to your system at t = 1s. Observe and plot how the controller brings the robot's tilt angle (θ) back to zero over time.

PART B

1- Modify or extend your model to more accurately capture the dynamics of your two-wheeled self-balancing robot.

2- Derive the governing equations for this improved model and express them in a form suitable for simulation (for example, by linearizing around the upright position).

3- Implement your improved model in Simulink and re-tune your controller gains as needed to achieve stable balancing.

Deliverables:

- A written report detailing your approach, methods, and results for each step in Parts A and B (maximum 4 pages)
- Your Simulink model files (.slx)